Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2016, Article ID 3795961, 11 pages
http://dx.doi.org/10.1155/2016/3795961

Research Article

A Parameter Estimation Method for Nonlinear Systems Based on Improved Boundary Chicken Swarm Optimization

Shaolong Chen,[1] Renyu Yang,[2] Renhuan Yang,[1] Liu Yang,[1] Xiuzeng Yang,[3] Chuangbiao Xu,[1] Baoguo Xu,[4] Huatao Zhang,[5] Yaosheng Lu,[1] and Weiping Liu[1]

[1] *College of Information Science and Technology, Jinan University, Guangzhou 510632, China*
[2] *School of Mechanical and Power Engineering, Guangdong Ocean University, Zhanjiang 524088, China*
[3] *Department of Physics and Electronic Engineering, Guangxi Normal University for Nationalities, Chongzuo 532200, China*
[4] *School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China*
[5] *Key Laboratory of Astronomical Optics & Technology, Nanjing Institute of Astronomical Optics & Technology, Chinese Academy of Sciences, Nanjing 210042, China*

Correspondence should be addressed to Renhuan Yang; tyangrh@jnu.edu.cn

Received 11 August 2016; Revised 12 November 2016; Accepted 20 November 2016

Academic Editor: Stefan Balint

Parameter estimation is an important problem in nonlinear system modeling and control. Through constructing an appropriate fitness function, parameter estimation of system could be converted to a multidimensional parameter optimization problem. As a novel swarm intelligence algorithm, chicken swarm optimization (CSO) has attracted much attention owing to its good global convergence and robustness. In this paper, a method based on improved boundary chicken swarm optimization (IBCSO) is proposed for parameter estimation of nonlinear systems, demonstrated and tested by Lorenz system and a coupling motor system. Furthermore, we have analyzed the influence of time series on the estimation accuracy. Computer simulation results show it is feasible and with desirable performance for parameter estimation of nonlinear systems.

1. Introduction

In the past decade, control and synchronization of nonlinear system in industry have attracted much attention. Some effective methods for nonlinear system control and synchronization are proposed and applied in engineering [1–10]. However, most of these methods are based on hypothesis that the parameters of system are known. They are generally inapplicable if system parameters are unknown. However, in practice, parameters of system are difficult to be known or measured due to complexity and unobservability of system. Therefore, parameter estimation is needed in modeling and control of these nonlinear systems.

Dynamic system identification is an inverse problem based on the input data and output data measured by experiment. After a mathematical model is established to reflect the essential characteristics of the system, parameters need to be identified. In general, the dynamics of nonlinear system in industry can be described by corresponding mathematical model. However, parameters need to be identified according to practice data, which is generally difficult. In the field of parameter estimation for nonlinear system, some effective methods have been proposed during the past few years. For instance, Gao and Hu [11] reported parameter estimation of chaotic system by using discontinuous drive signals. Blanchard et al. [12] proposed a parameter estimation approach that uses polynomial chaos to propagate uncertainties, estimating error covariance in the extended Kalman filter framework. Liu et al. [13] presented a method for estimating one-dimensional discrete chaotic system based on mean value function. In addition, some intelligent optimization algorithms have been proposed for the parameter identification, such as genetic algorithm (GA) [14], particle swarm optimization (PSO) [15–18], differential evolution (DE) [19], ant swarm optimization algorithm (AS) [20], bat algorithm (BA) [21], cuckoo search optimization algorithm

(CS) [22], and teaching learning based optimization (TLBO) [23]. However, research on the influence of time series on the estimation accuracy of multidimensional nonlinear system is rare.

Recently, a new bioinspired optimization algorithm, namely, chicken swarm optimization (CSO) [24] is proposed, and it mimics the hierarchy and behavior of chickens. The algorithm proved to be very promising and could outperform existing algorithms such as PSO, DE, and BA [24]. Due to the excellent global convergence and robustness, CSO has been widely applied in engineering [25, 26]. Similar to other bioinspired optimization algorithms, the CSO algorithm can be further improved to enhance convergence speed and convergence precision. In this paper, parameter estimation of nonlinear system is transformed into a multidimensional parameter optimization problem by constructing an appropriate fitness function, and then a method based on improved boundary chicken swarm optimization (IBCSO) is proposed for the multidimensional parameters optimization problem. However, to our best knowledge, there is still not research work applying chicken swarm optimization to solve parameter estimation problem of nonlinear system in previous literatures. Furthermore, we have analyzed the influence of time series on the estimation accuracy. We demonstrated and tested the proposed method by Lorenz nonlinear system [16] and coupling motor system [27]. Computer simulation results show the proposed method is feasible with desirable performance for parameter estimation of nonlinear systems.

This paper is organized as follows. The problem formulation is briefly addressed in Section 2. In Section 3, we proposed an improved boundary chicken swarm optimization. In Section 4, we analyze the influence of time series on the estimation accuracy. Computer simulation results are presented in Section 5. Section 6 is the conclusion.

2. Problem Description

A general nonlinear system can be described by the following equation:

$$\dot{X} = F\left(X, X_0, \mu_0\right). \tag{1}$$

Here, $X = (X_1, X_2, \dots, X_n)^T \in R^n$ represents the state vector of the original system. X_0 is the initial value of the system. $\mu_0 = (\mu_{10}, \mu_{20}, \dots, \mu_{d0})^T$ are the true value of the parameters of the system.

Assume that the structure of system (1) is known; thus, the estimated system can be written as

$$\dot{Y} = F\left(Y, Y_0, \mu\right). \tag{2}$$

Here, $Y = (Y_1, Y_2, \dots, Y_n)^T \in R^n$ represents the state vector of the estimated system. Y_0 is the initial value of the system, and $Y_0 = X_0$. $\mu = (\mu_1, \mu_2, \dots, \mu_d)^T$ are the estimated value of the system parameters.

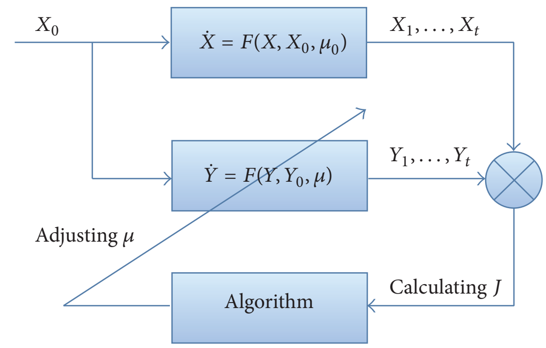

Figure 1: The principle of parameter estimation for a nonlinear system.

Based on hereinbefore analysis, the parameter estimation problem can be transformed into the following optimization problem:

$$J = \frac{1}{L} * \sum_{t=1}^{L} \left\| X_t - Y_t \right\|. \tag{3}$$

Here, L denotes the time series. X_t and Y_t coordinates represent the states of the original system and the estimated system at time t, respectively.

The parameter estimation of the nonlinear system can be formulated into multidimensional parameters optimization problem, where the decision vector is μ and the optimization goal is to minimize J. The principle of parameter estimation for nonlinear system from an optimizing perspective is shown in Figure 1.

It is difficult to estimate parameters of nonlinear system due to complexity and unobservability of system, so it is challenging to approach satisfactory result by using traditional optimization methods. Therefore, an improved boundary chicken swarm optimization (IBCSO) is proposed to develop an effective parameter estimation method for nonlinear systems in this paper.

3. Improved Boundary Chicken Swarm Optimization

3.1. Chicken Swarm Optimization. Chicken swarm optimization (CSO) is a novel swarm intelligence algorithm, which simulates the hierarchy and behavior of chickens. In this algorithm, the chickens were divided into several groups, each of which consists of one rooster and many hens and chicks. Assume N_R, N_H, N_C, and N_M denote the number of the roosters, the hens, the chicks, and the mother hens, respectively. The best N_R chickens would be assumed to be roosters, while the worst N_C ones would be regarded as chicks, and the rest are treated as hens. All N virtual chickens, depicted by their positions $x_{i,j}^t$ $(i \in [1, N],\ j \in [1, D])$ at time step t, search for food in a D-dimensional space. $px_{i,j}$ $(i \in [1, N],\ j \in [1, D])$ represent the optimal position of ith now [24].

Different chickens follow different laws of motions. The roosters with better fitness values have priority for food access

Table 1: Estimation results for different time series L.

		θ_1	θ_2	θ_3	J
$L = 10$	Best result	10.000000	28.000000	2.666667	$2.595820e-14$
	Worst result	9.999944	27.999901	2.666661	$1.832950e-10$
	Average result	9.999991	27.999982	2.666666	$2.115565e-11$
	Standard deviation	$1.274592e-5$	$2.162007e-5$	$1.324675e-6$	$3.900723e-11$
$L = 100$	Best result	10.000000	27.999999	2.666666	$2.370576e-11$
	Worst result	9.999854	27.999859	2.666610	$7.786259e-8$
	Average result	9.999976	27.999970	2.666656	$7.910590e-9$
	Standard deviation	$2.783841e-5$	$3.589249e-5$	$1.066062e-5$	$1.362798e-8$
$L = 200$	Best result	9.999990	27.999997	2.666666	$1.264891e-9$
	Worst result	9.872441	27.934373	2.623538	0.054542
	Average result	9.996019	27.998028	2.665363	0.001121
	Standard deviation	0.018049	0.009255	0.006086	0.007710
$L = 500$	Best result	9.997352	27.998694	2.665944	$1.850371e-05$
	Worst result	9.138323	27.542075	2.353588	2.743463
	Average result	9.712258	27.860182	2.586612	0.378123
	Standard deviation	0.220905	0.104500	0.068190	0.587383

than the ones with worse fitness values, and location update formula is as follows:

$$x_{i,j}^t = px_{i,j} * \left(1 + \text{randn}\left(0, \sigma^2\right)\right), \quad (4)$$

$$\sigma^2 = \begin{cases} 1, & \text{if } f_i \le f_k, \\ \exp\left(\dfrac{f_k - f_i}{|f_i| + \epsilon}\right), & \text{otherwise.} \end{cases} \quad (5)$$

Here, $\text{randn}(0, \sigma^2)$ is a Gaussian distribution with mean 0 and standard deviation σ^2. ϵ is the smallest constant in the computer. k is randomly selected from the roosters group, and $k \ne i$. f is fitness value of corresponding x.

The hen's location update formula is as follows:

$$x_{i,j}^t = px_{i,j} + S_1 * \text{rand} * \left(px_{r1,j} - px_{i,j}\right) + S_2 * \text{rand} * \left(px_{r2,j} - px_{i,j}\right), \quad (6)$$

$$S_1 = \exp\left(\frac{f_i - f_{r1}}{\text{abs}(f_i) + \epsilon}\right), \quad (7)$$

$$S_2 = \exp\left(f_{r2} - f_i\right). \quad (8)$$

Here, rand is a uniform random distribution of $[0, 1]$. $r1$ is the ith hen's group-mate, $r2$ is randomly chosen from the swarm, and $r1 \ne r2$.

The chicks location update formula is as follows:

$$x_{i,j}^t = px_{i,j} + FL * \left(px_{m,j} - px_{i,j}\right). \quad (9)$$

Here, m is the ith chick's mother. FL is a uniform random distribution of $[0, 2]$.

if $x_{i,j}^t < Lb_j$
  $x_{i,j}^t = Lb_j$;
else if $x_{i,j}^t > Ub_j$
  $x_{i,j}^t = Ub_j$;
end if

Algorithm 1: Cross-border processing function.

if $x_{i,j}^t < Lb_j \parallel x_{i,j}^t > Ub_j$
  $w = 0.4 * |px_{\text{best},j} - px_{i,j}|$;
  $\text{temp} = px_{\text{best},j} + w * \text{randn}(0, 1)$;
  if $Lb_j \le \text{temp} \le Ub_j$
    $x_{i,j}^t = \text{temp}$;
  else
    $x_{i,j}^t = px_{i,j}$;
  end if
end if

Algorithm 2: Improved cross-border processing function.

3.2. Improved Boundary Chicken Swarm Optimization. In the standard chicken swarm optimization algorithm, when a component goes cross the border, it is then replaced with a corresponding value of upper and lower boundary, and the function of cross-border processing is shown in Algorithm 1. In this paper, in order to improve the convergence speed and convergence precision of the CSO, we proposed an improved boundary chicken swarm optimization (IBCSO); when a component goes cross the border, it is then replaced with a random component between the similar component of the individual's best solution and the global best solution so far, and the function of improved cross-border processing is shown in Algorithm 2. Therefore, we get the process of improved boundary chicken swarm optimization shown in Algorithm 3.

```
Initialize a population of N chickens and define the related parameters;
Evaluate the fitness values for each individual, set current each individual's
 position and fitness value, and set the current global best individual's position
 and fitness value;
for t = 1 to M
   if t % G == 1
      Rank the chickens' fitness values and establish a hierarchal order in the swarm;
      Divide the swarm into different groups, and determine the relationship
       between the chicks and mother hens in a group;
   end if
   Rank the chickens' fitness values;
   for i = 1 to N
      if i == rooster    Update its location using equation (4);    end if
      if i == hen    Update its location using equation (6);    end if
      if i == chick    Update its location using equation (9);    end if
      Improved cross-border processing function;
      Evaluate the fitness values for i;
      If the new fitness value is better than the current individual's fitness value,
       update the individual's position and fitness value;
      If the new fitness value is better than the current global best individual's
       fitness value, then update the current global best individual's position
       and fitness value;
      If a stopping criterion is met, then output the current global best
       individual's position and fitness value;
   end for
end for
```

Algorithm 3: Improved boundary chicken swarm optimization.

Table 2: Statistical results from the IBCSO, CSO, PSO, GA, and TLBO.

Algorithms		θ_1	θ_2	θ_3	J
IBCSO	Best result	10.000000	28.000000	2.666667	$1.311671e-14$
	Worst result	9.999950	27.999993	2.666662	$9.852850e-11$
	Average result	9.999994	27.999998	2.666666	$6.801939e-12$
	Standard deviation	$7.903017e-6$	$1.629054e-6$	$7.196902e-7$	$1.588317e-11$
CSO	Best result	9.999997	27.999997	2.666667	$9.164669e-11$
	Worst result	9.999600	27.999761	2.666606	$1.395802e-8$
	Average result	9.999875	27.999942	2.666653	$2.939848e-9$
	Standard deviation	$9.412609e-5$	$5.409299e-5$	$1.249644e-5$	$3.209969e-9$
PSO	Best result	9.999725	27.999973	2.666661	$1.539265e-8$
	Worst result	9	27.913029	2.660311	0.041631
	Average result	9.798000	27.984713	2.664953	0.007229
	Standard deviation	0.395116	0.026362	0.001990	0.014707
GA	Best result	9.987552	27.999037	2.665611	0.000427
	Worst result	9.402872	27.743833	2.591086	0.018260
	Average result	9.829154	27.911747	2.642162	0.006100
	Standard deviation	0.145217	0.057552	0.018232	0.004753
TLBO	Best result	9.999999	28.000000	2.666666	$6.278703e-12$
	Worst result	9.999840	27.999899	2.666642	$2.610822e-9$
	Average result	9.999951	27.999976	2.666661	$4.760403e-10$
	Standard deviation	$4.446187e-05$	$1.892609e-5$	$5.115130e-06$	$5.633741e-10$

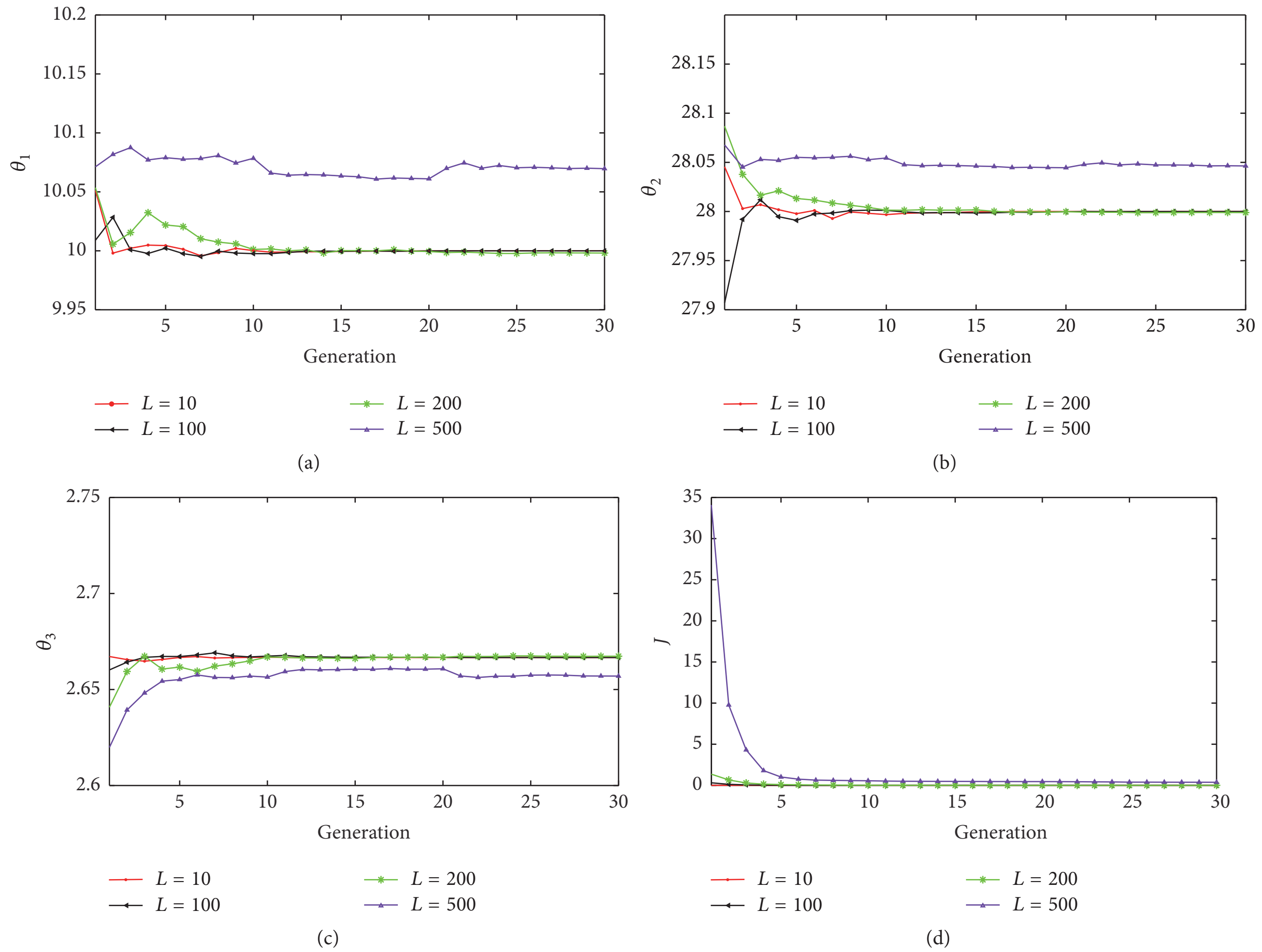

FIGURE 2: The evolving process of the average values for different time series L. (a) θ_1; (b) θ_2; (c) θ_3; (d) J.

4. Estimation Accuracy Analysis for a Nonlinear System Example

In this section, in order to discuss the influence of the time series on the estimation accuracy, we consider a Lorenz system.

$$\begin{aligned}\dot{X} &= \theta_1 * (Y - X), \\ \dot{Y} &= \theta_2 * X - X * Z - Y, \\ \dot{Z} &= X * Y - \theta_3 * Z.\end{aligned} \qquad (10)$$

Here, X, Y, and Z are the state variables; $\theta_1 = 10$, $\theta_2 = 28$, and $\theta_3 = 8/3$ are the original parameters.

We initialize system (10) with a state x_0, which is randomly selected from the evolution process of the Lorenz system. The searching ranges are set as follows: $9 < \theta_1 < 11$, $20 < \theta_2 < 30$, and $2 < \theta_3 < 3$. The population size and maximum cycle number are set to be $N = 60$, $M = 30$. The parameters of the IBCSO are configured as follows: $N_R = 0.2N$, $N_H = 0.6N$, $N_C = 0.2N$, $N_M = 0.1N$, $G = 10$, and $FL \in [0.5, 0.9]$ [24]. Let time series L be different values and run the program of improved boundary chicken swarm optimization algorithm; we use the IBCSO algorithm to estimate the unknown parameters. To make a fair comparison, all cases are run 50 times, and the initial population is set as uniform same value for all the time series L at the same time run. Table 1 lists the estimation results for different time series L. The evolving processes of the average values for different time series L are shown in Figure 2.

Seen from Table 1, the estimation accuracy declines as L increases. In addition, Figure 2 once again shows that estimation accuracy declines as time series L increases. The reason is that the critical sensitivity of the nonlinear system to initial conditions and parameters results in that the objective function becomes very complicated as the increment of L.

5. Parameters Estimation Results for Nonlinear Systems and Discussions

5.1. Lorenz System

5.1.1. Offline Estimation. In this simulation, system (10) is used to test the performance of the IBCSO compared with

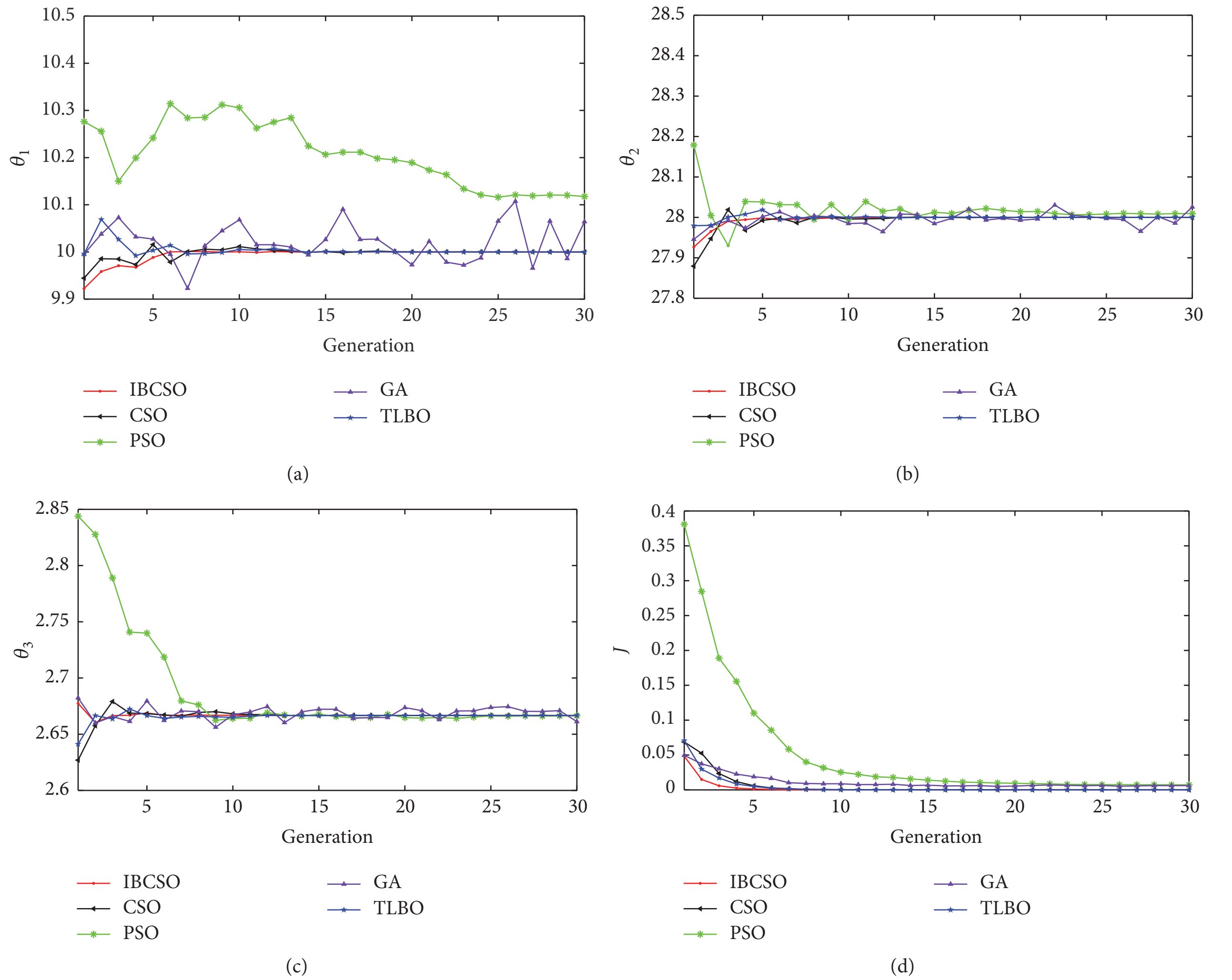

FIGURE 3: The evolving process of the average values obtained by IBCSO, CSO, PSO, GA, and TLBO in offline mode. (a) θ_1; (b) θ_2; (c) θ_3; (d) J.

that of CSO, PSO, GA, and TLBO. We initialize the system with a state x_0, which is randomly selected from the evolution process of the system. The searching ranges, population size, maximum cycle number, and time series for IBCSO, CSO, PSO, GA, and TLBO are all set as follows: $9 < \theta_1 < 11$, $20 < \theta_2 < 30$, $2 < \theta_3 < 3$, $N = 60$, $M = 30$, and $L = 10$. The parameters of the algorithms are configured as follows. For IBCSO and CSO, $N_R = 0.2N$, $N_H = 0.6N$, $N_C = 0.2N$, $N_M = 0.1N$, $G = 2$, and $FL \in [0.5, 0.9]$ [24]. For PSO and GA, all the parameters are the same as those used in literature [16]. For TLBO, all the parameters are the same as those used in literature [23]. To make a fair comparison, all algorithms are run 50 times, and the initial population is set as uniform same value for all the optimization algorithms at the same time run. Table 2 lists results obtained by IBCSO, CSO, PSO, GA, and TLBO. The evolving processes of the average values obtained by IBCSO, CSO, PSO, GA, and TLBO are shown in Figure 3. Moreover, to compare the iteration number of the algorithms, $J \le 10^{-10}$ is considered as the stopping criteria. The maximum cycle number is set to 1000, and other conditions are the same as above. Table 3 lists the results obtained by IBCSO, CSO, PSO, GA, and TLBO.

5.1.2. Online Estimation. In this simulation, we investigate the capability of the algorithms in chasing the alternations in the parameters of the system. In the first part, $\theta_1 = 10$, $\theta_2 = 28$, and $\theta_3 = 8/3$. In the second part, θ_1 moves down to 9.5 from 10, θ_2 moves down to 27 from 28, and θ_3 moves down to 2.6 from 8/3 in the 31st iteration. The maximum cycle number is set to 60, and the others conditions in this part are the same as the conditions indicated in the offline mode. The estimation of online parameters of the system can be seen in Figure 4.

5.2. Coupling Motor System. In this section, in order to further prove the performance of the proposed method, we consider a coupling motor system [27].

$$\begin{aligned}\dot{x} &= -\theta_2 x + y\left(z + \theta_1\right),\\ \dot{y} &= -\theta_2 y + x\left(z - \theta_1\right),\\ \dot{z} &= \theta_3 z - xy.\end{aligned} \tag{11}$$

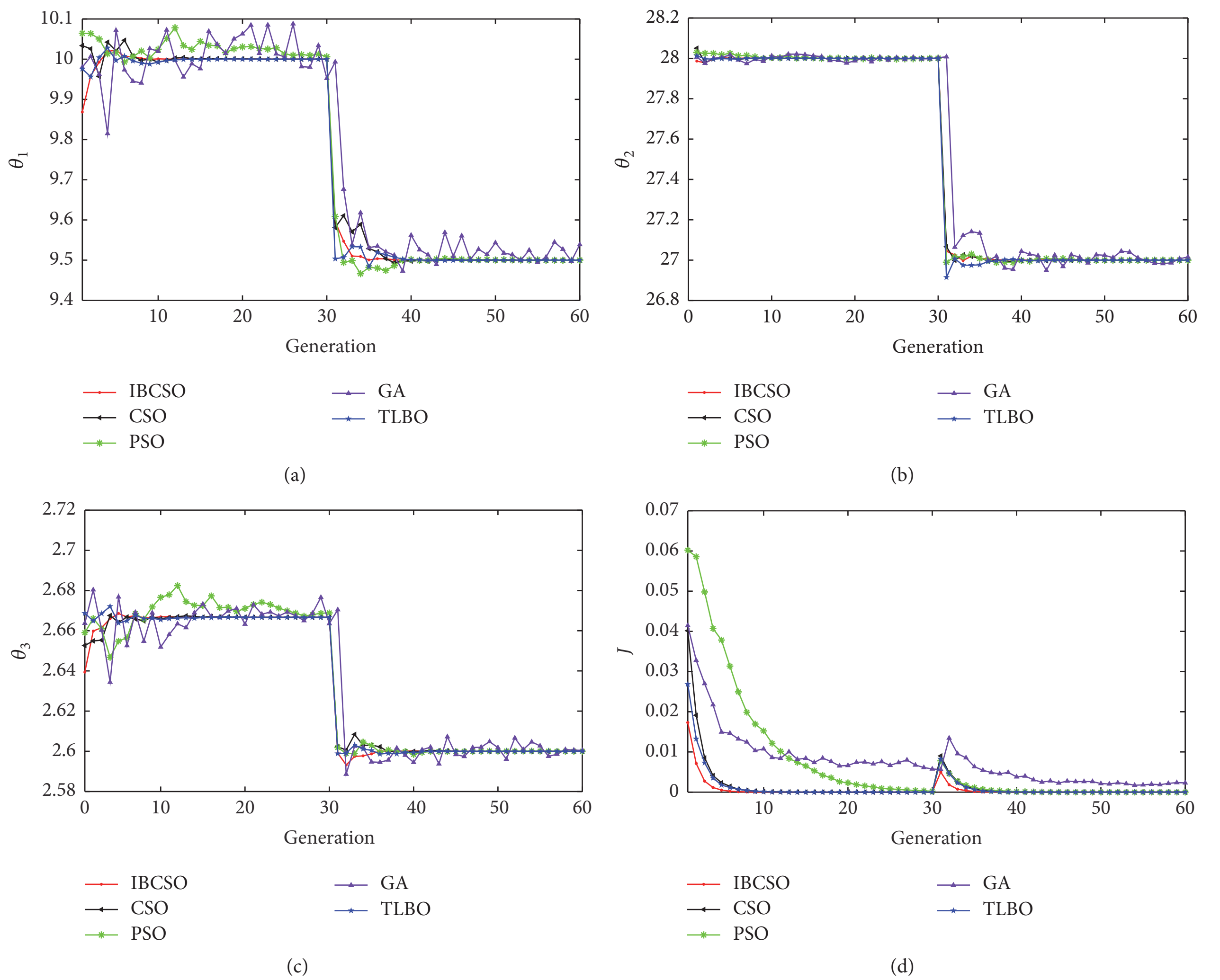

Figure 4: The evolving process of the average values obtained by IBCSO, CSO, PSO, GA, and TLBO in online mode. (a) θ_1; (b) θ_2; (c) θ_3; (d) J.

Table 3: Iterations required by IBCSO, CSO, PSO, GA, and TLBO.

Algorithms		Iterations
IBCSO	Best result	19
	Worst result	29
	Average result	24
PSO	Best result	183
	Worst result	1000
	Average result	481
TLBO	Best result	24
	Worst result	32
	Average result	29
CSO	Best result	27
	Worst result	37
	Average result	32
GA	Best result	1000
	Worst result	1000
	Average result	1000

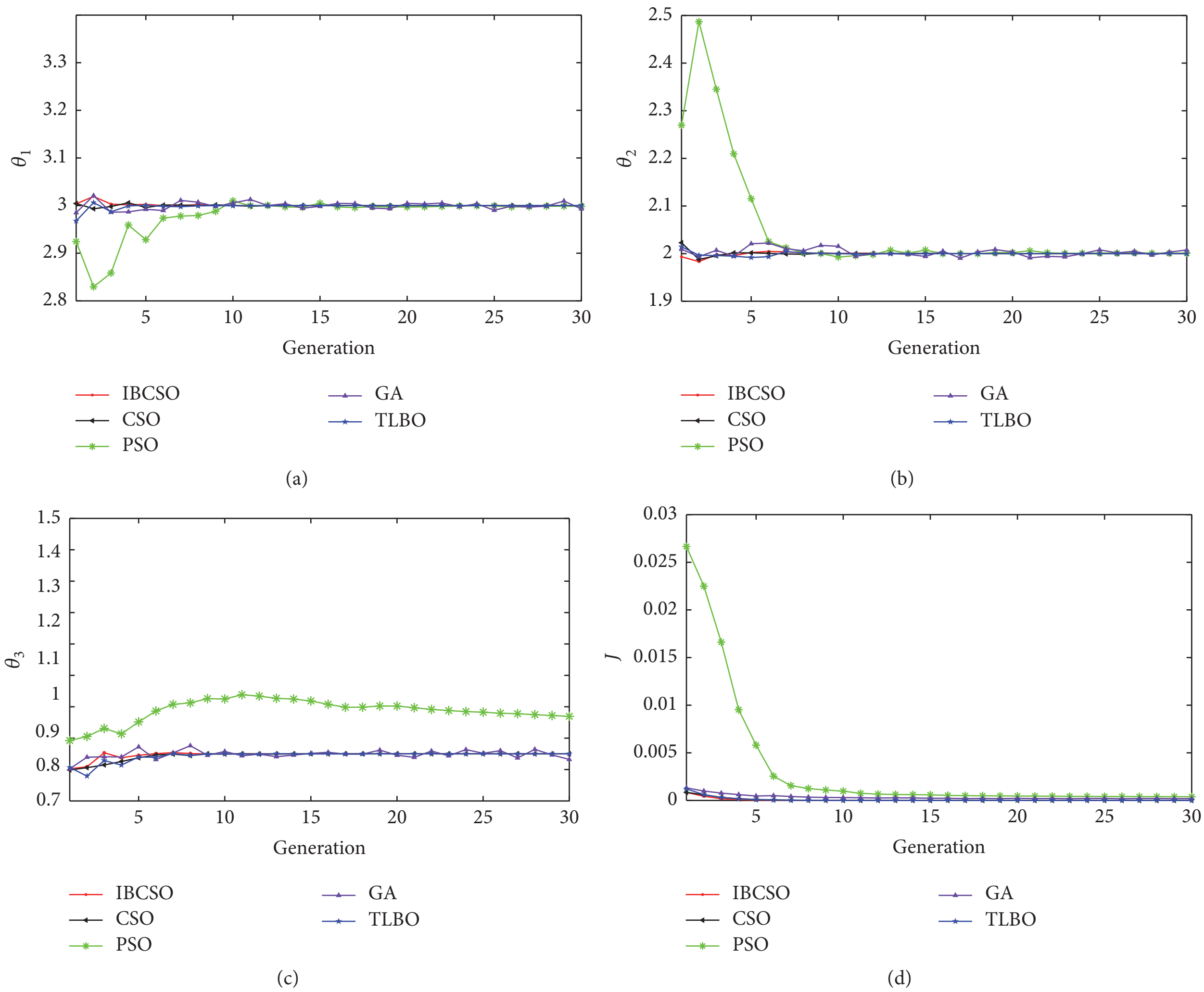

Figure 5: The evolving process of the average values obtained by IBCSO, CSO, PSO, GA, and TLBO in offline mode. (a) θ_1; (b) θ_2; (c) θ_3; (d) J.

Here, x, y, and z are the state variables; $\theta_1 = 3$, $\theta_2 = 2$, and $\theta_3 = 0.75$ are the original parameters.

5.2.1. Offline Estimation. In this simulation, system (11) is used to test the performance of the IBCSO compared with that of CSO, PSO, GA, and TLBO. We initialize the system with a state x_0, which is randomly selected from the evolution process of the system. The searching ranges, population size, maximum cycle number, and time series for IBCSO, CSO, PSO, GA, and TLBO are all set as follows: $2 < \theta_1 < 4$, $1 < \theta_2 < 3$, $0 < \theta_3 < 1$, $N = 60$, $M = 30$, and $L = 10$. The parameters of the algorithms are configured as follows. For IBCSO and CSO, $N_R = 0.2N$, $N_H = 0.6N$, $N_C = 0.2N$, $N_M = 0.1N$, $G = 2$, and $FL \in [0.5, 0.9]$ [24]. For PSO and GA, all the parameters are the same as those used in literature [16]. For TLBO, all the parameters are the same as those used in literature [23]. To make a fair comparison, all algorithms are run 50 times, and the initial population is set as uniform same value for all the optimization algorithms at the same time run. Table 4 lists the results obtained by IBCSO, CSO, PSO, GA, and TLBO. The evolving processes of the average values obtained by IBCSO, CSO, PSO, GA, and TLBO are shown in Figure 5. Moreover, to compare the iteration number of the algorithms, $J \leq 10^{-10}$ is considered as the stopping criteria. The maximum cycle number is set to 1000, and other conditions are the same as above. Table 5 lists the results obtained by IBCSO, CSO, PSO, GA, and TLBO.

5.2.2. Online Estimation. In this simulation, we investigate the capability of the algorithms in chasing the alternations in the parameters of the system. In the first part, $\theta_1 = 3$, $\theta_2 = 2$, and $\theta_3 = 0.75$. In the second part, θ_1 moves down to 2.5 from 3, θ_2 moves down to 2.5 from 2, and θ_3 moves down to 0.5 from 0.75 in the 31st iteration. The maximum cycle number is set to 60, and the other conditions in this part are the same as the conditions indicated in the offline mode. The estimation of online parameters of the system can be seen in Figure 6.

From the above two examples, the results presented demonstrate that a good optimal performance can be achieved by the proposed IBCSO algorithm. As shown in

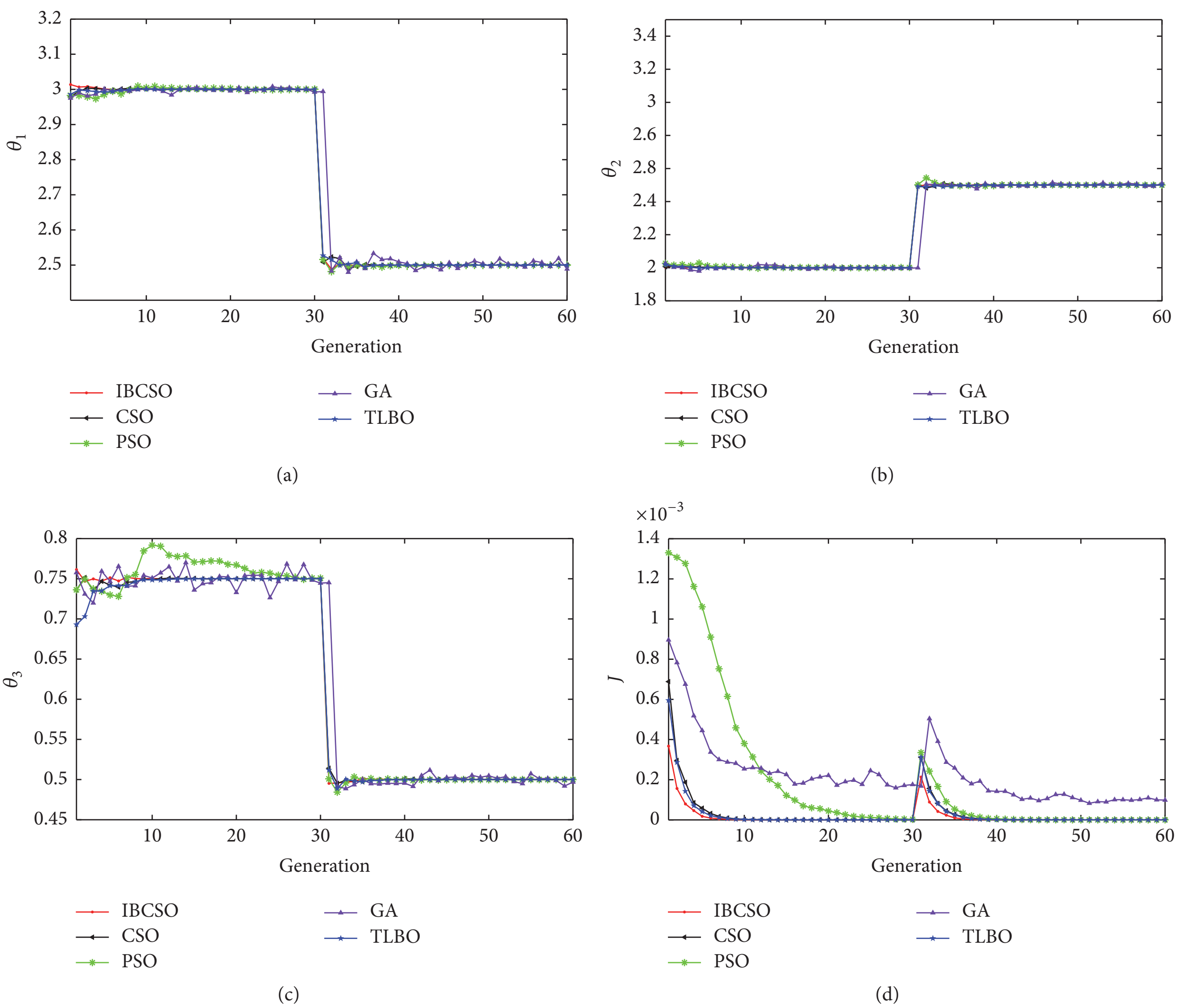

FIGURE 6: The evolving process of the average values obtained by IBCSO, CSO, PSO, GA, and TLBO in online mode. (a) θ_1; (b) θ_2; (c) θ_3; (d) J.

Tables 2 and 4, the best, the average, the worst results and standard deviation obtained by IBCSO are all better than those obtained by CSO, PSO, GA, and TLBO, respectively. In addition, Figures 3 and 5 once again show that IBCSO is of better performance than CSO, PSO, GA, and TLBO in terms of convergence speed and convergence precision. Moreover, from Tables 3 and 5, it is confirmed that the IBCSO spends less iterations to reach a predefined threshold compared with CSO, PSO, GA, and TLBO. Furthermore, as shown in Figures 4 and 6, tracking the changes of the system parameters by the IBCSO is well-performed.

6. Conclusion

In this paper, a method based on improved boundary chicken swarm optimization (IBCSO) algorithm is proposed to solve the problem of parameter estimation for nonlinear systems. Computer simulation based on two nonlinear systems examples and comparisons with results obtained by CSO, PSO, GA, and TLBO demonstrated the effectiveness of the proposed method. Furthermore, we have analyzed the influence of time series on the estimation accuracy. According to theoretical analysis and computer simulation, we achieved the following conclusions: shorter length of time series will benefit the estimation accuracy because that longer time series will make the objective function complicated. Therefore, it is very important to select a suitable time series to reduce the estimation bias of aim nonlinear systems. Although it is demonstrated by two nonlinear systems examples in this paper, the proposed method can also be used as a promising tool for numerical optimization problems in engineering.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Table 4: Statistical results from the IBCSO, CSO, PSO, GA, and TLBO.

Algorithms		θ_1	θ_2	θ_3	J
IBCSO	Best result	3.000000	2.000000	0.750000	$1.515880e-14$
	Worst result	2.999988	1.999984	0.749954	$2.955687e-11$
	Average result	2.999996	1.999994	0.749992	$3.818720e-12$
	Standard deviation	$3.528640e-6$	$4.394600e-6$	$7.615395e-7$	$4.837061e-12$
CSO	Best result	2.999998	2.000000	0.750000	$1.089261e-12$
	Worst result	2.999933	1.999945	0.749913	$1.445733e-10$
	Average result	2.999984	1.999982	0.749976	$3.833475e-11$
	Standard deviation	$1.202585e-5$	$1.400733e-5$	$2.199347e-5$	$3.785071e-11$
PSO	Best result	2.999971	1.999995	0.749776	$1.415251e-8$
	Worst result	2.981136	1.984547	0.500000	0.000812
	Average result	2.997187	1.997420	0.628964	0.000389
	Standard deviation	0.003126	0.002847	0.124597	0.000407
GA	Best result	2.998662	1.999492	0.746170	$1.387276e-05$
	Worst result	2.880957	1.883540	0.590241	0.000495
	Average result	2.960151	1.962746	0.696085	0.000188
	Standard deviation	0.030768	0.028565	0.038388	0.000136
TLBO	Best result	3.000000	2.000000	0.750000	$6.668050e-14$
	Worst result	2.999974	1.999982	0.749947	$3.784144e-11$
	Average result	2.999994	1.999995	0.749991	$5.022291e-12$
	Standard deviation	$4.982352e-06$	$4.686890e-6$	$9.583826e-06$	$6.587053e-12$

Table 5: Iterations required by IBCSO, CSO, PSO, GA, and TLBO.

Algorithms		Iterations
IBCSO	Best result	22
	Worst result	29
	Average result	26
PSO	Best result	175
	Worst result	1000
	Average result	450
TLBO	Best result	23
	Worst result	31
	Average result	27
CSO	Best result	26
	Worst result	34
	Average result	30
GA	Best result	1000
	Worst result	1000
	Average result	1000

Acknowledgments

This work was supported in part by the NSF project of China (nos. 61302131, 61673114, and 61104206), the Fundamental Research Funds for the Central Universities (no. 2242016R30025), project of International S & T Cooperation Program of China (ISTCP) (no. 2015DFI12970), projects of Guangdong Science and Technology Program (nos. 2015B010105012, 2014B050505011, 2013B010136002, 2015B020214004, 2014A050503046, and 2015B020233010), project of Guangzhou Science and Technology Program (no. 201508020083), and project of Guangxi Science and Technology Program (no. 2015AA08210).

References

[1] J. Lai, H. Zhou, W. Hu, X. Lu, and L. Zhong, "Synchronization of hybrid microgrids with communication latency," *Mathematical Problems in Engineering*, vol. 2015, Article ID 586260, 10 pages, 2015.

[2] A. Alfi, A. A. Kalat, and M. H. Khooban, "Adaptive fuzzy sliding mode control for synchronization of uncertain non-identical

chaotic systems using bacterial foraging optimization," *Journal of Intelligent & Fuzzy Systems*, vol. 26, no. 5, pp. 2567–2576, 2014.

[3] M. R. Khalghani, M. A. Shamsi-nejad, and M. H. Khooban, "Dynamic voltage restorer control using bi-objective optimisation to improve power quality's indices," *IET Science, Measurement & Technology*, vol. 8, no. 4, pp. 203–213, 2014.

[4] M. H. Khooban, D. N. M. Abadi, A. Alfi, and M. Siahi, "Optimal type-2 fuzzy controller for HVAC systems," *Automatika*, vol. 55, no. 1, pp. 69–78, 2014.

[5] T. Hinze, M. Schumann, C. Bodenstein, I. Heiland, and S. Schuster, "Biochemical frequency control by synchronisation of coupled repressilators: an in Silico study of modules for circadian clock systems," *Computational Intelligence and Neuroscience*, vol. 2011, Article ID 262189, 9 pages, 2011.

[6] M. H. Khooban, T. Niknam, and M. Sha-Sadeghi, "Speed control of electrical vehicles: a time-varying proportional–integral controller-based type-2 fuzzy logic," *IET Science, Measurement & Technology*, vol. 10, no. 3, pp. 185–192, 2016.

[7] M. R. Soltanpour, M. H. Khooban, and M. R. Khalghani, "An optimal and intelligent control strategy for a class of nonlinear systems: adaptive fuzzy sliding mode," *Journal of Vibration and Control*, vol. 22, no. 1, pp. 159–175, 2016.

[8] H.-T. Yau, Y.-C. Pu, and S. C. Li, "An FPGA-based PID controller design for chaos synchronization by evolutionary programming," *Discrete Dynamics in Nature and Society*, vol. 2011, Article ID 516031, 11 pages, 2011.

[9] R. Yang and A. Song, "Effect of positive feedback with threshold control on stochastic resonance of bi-stable systems," *International Journal of Modern Physics B*, vol. 26, no. 3, Article ID 1250019, 10 pages, 2012.

[10] R. Yang, A. Song, and W. Yuan, "Enhancement of spike synchrony in HindmarshRose neural networks by randomly rewiring connections," *Modern Physics Letters B*, vol. 23, no. 11, pp. 1405–1414, 2009.

[11] X. Gao and H. Hu, "Adaptive-impulsive synchronization and parameters estimation of chaotic systems with unknown parameters by using discontinuous drive signals," *Applied Mathematical Modelling*, vol. 39, no. 14, pp. 3980–3989, 2015.

[12] E. D. Blanchard, A. Sandu, and C. Sandu, "A polynomial chaos-based kalman filter approach for parameter estimation of mechanical systems," *Journal of Dynamic Systems, Measurement and Control*, vol. 132, no. 6, Article ID 061404, 2010.

[13] L. Liu, J. Hu, H. Li, J. Li, Z. He, and C. Han, "Parameter estimation of a class one-dimensional discrete chaotic system," *Discrete Dynamics in Nature and Society*, vol. 2011, Article ID 696017, 9 pages, 2011.

[14] C. Tao, Y. Zhang, and J. J. Jiang, "Estimating system parameters from chaotic time series with synchronization optimized by a genetic algorithm," *Physical Review E*, vol. 76, no. 1, Article ID 016209, 2007.

[15] D. Selişteanu, D. Şendrescu, V. Georgeanu, and M. Roman, "Mammalian cell culture process for monoclonal antibody production: nonlinear modelling and parameter estimation," *BioMed Research International*, vol. 2015, Article ID 598721, 16 pages, 2015.

[16] Q. He, L. Wang, and B. Liu, "Parameter estimation for chaotic systems by particle swarm optimization," *Chaos, Solitons and Fractals*, vol. 34, no. 2, pp. 654–661, 2007.

[17] M. Jakubcová, P. Máca, and P. Pech, "Parameter estimation in rainfall-runoff modelling using distributed versions of particle swarm optimization algorithm," *Mathematical Problems in Engineering*, vol. 2015, Article ID 968067, 13 pages, 2015.

[18] A. Alfi, "PSO with adaptive mutation and inertia weight and its application in parameter estimation of dynamic systems," *Acta Automatica Sinica*, vol. 37, no. 5, pp. 541–549, 2011.

[19] W. Xiang, X. Meng, and M. An, "An alternate iterative differential evolution algorithm for parameter identification of chaotic systems," *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 740721, 11 pages, 2015.

[20] H. Peng, L. Li, Y. Yang, and F. Liu, "Parameter estimation of dynamical systems via a chaotic ant swarm," *Physical Review E*, vol. 81, no. 1, Article ID 016207, 2010.

[21] A. Rahimi, F. Bavafa, S. Aghababaei, M. H. Khooban, and S. V. Naghavi, "The online parameter identification of chaotic behaviour in permanent magnet synchronous motor by Self-Adaptive Learning Bat-inspired algorithm," *International Journal of Electrical Power & Energy Systems*, vol. 78, pp. 285–291, 2016.

[22] J. Wang, B. Zhou, and S. Zhou, "An improved cuckoo search optimization algorithm for the problem of chaotic systems parameter estimation," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 2959370, 8 pages, 2016.

[23] X. Chen, K. Yu, W. Du, W. Zhao, and G. Liu, "Parameters identification of solar cell models using generalized oppositional teaching learning based optimization," *Energy*, vol. 99, pp. 170–180, 2016.

[24] X. B. Meng, L. Yu, X. Z. Gao, and H. Z. Zhang, "A new bio-inspired algorithm: chicken swarm optimization," in *Advances in Swarm Intelligence*, vol. 8794 of *Lecture Notes in Computer Science*, pp. 86–94, Springer International, 2014.

[25] Y. L. Chen, P. L. He, and Y. H. Zhang, "Combining penalty function with modified chicken swarm optimization for constrained optimization," *Advances in Intelligent Systems Research*, vol. 126, pp. 1899–1907, 2015.

[26] P. Chen and Y. Y. Mao, "Wireless sensor network node localization algorithm based on chicken swarm optimization and multi-power mobile anchor," *AER-Advances in Engineering Research*, vol. 67, pp. 245–250, 2016.

[27] J.-H. Hao and N.-Y. Sun, "The characteristics of the chaotic parameters for a loss type of modified coupled dynamic system," *Wuli Xuebao/Acta Physica Sinica*, vol. 61, no. 15, Article ID 150504, 2012.